AMENDMENT TO THE CLAIMS

1. (ORIGINAL) A computer-readable medium having computer-executable instructions for performing steps for building a symbol table for storing sort weights for a plurality of linguistic symbols used in a plurality of languages supported by a computer system, comprising:

constructing the symbol table to contain a list of code points each uniquely identifying one of the symbols, and a sort weight for the symbol identified by said each code point;

providing a plurality of compression tables, each compression table pertaining to one of the supported languages and having a compression type and containing compressions of symbols of that compression type;

for each code point in the symbol table, sorting the compression tables to identify a highest compression type our compressions beginning with the symbol identified by said each code point; and

storing in the symbol table a tag for each code point to indicate said highest compression type for said each code point.

2. (ORIGINAL) A computer-readable medium as in claim 1, wherein the code points are assigned to the symbols according to the Unicode standard.

3. (ORIGINAL) A computer-readable medium as in claim 1, wherein the tag for each code point is stored as a portion of the sort weight of the symbol identified by said each code point.

4. (ORIGINAL) A computer-readable medium as in claim 3, wherein the sort weight of the symbol identified by said each code point comprises a case weight value, and wherein the tag for said each code point is stored as part of the case weight value for said each code point.

5. (ORIGINAL) A computer-readable medium as in claim 1, further comprising computer-executable instructions for performing steps of sorting compressions in each of the compression tables based on combinations of code points of the compressions in said each compression table.

6. (ORIGINAL) A method of building a symbol table for storing sort weights for a plurality of linguistic symbols used in a plurality of languages supported by a computer system, comprising:

constructing the symbol table to contain a list of code points each uniquely identifying one of the symbols, and a sort weight for the symbol identified by said each code point;

providing a plurality of compression tables, each compression table pertaining to one of the supported languages and having a compression type and containing compressions of symbols of that compression type;

for each code point in the symbol table, sorting the compression tables to identify a highest compression type for compressions beginning with the symbol identified by said each code point; and

storing a tag in the symbol table for each code point to indicate said highest compression type for said each code point.

7. (ORIGINAL) A method as in claim 6, wherein the code points are assigned to the symbols according to the Unicode standard.

8. (ORIGINAL) A method as in claim 6, wherein the tag for each code point is stored as a portion of the sort weight of the symbol identified by said each code point.

9. (ORIGINAL) A method as in claim 8, wherein the sort weight of the symbol identified by said each code point comprises a case weight value, and wherein the tag for said each code point is stored as part of the case weight value for said each code point.

10. (ORIGINAL) A method as in claim 6, further including the step of sorting compressions in each of the compression tables based on combinations of code points of the compressions in said each compression table.

11. (ORIGINAL) A computer-readable medium having computer-executable instructions for performing steps for a computer search program to carry out a linguistic sorting operation, comprising:

receiving an input string containing a plurality of linguistic symbols used in a given language;

for a first symbol in a combination of symbols in the input string, referencing a symbol table to obtain a highest compression type for compressions beginning with said first symbol, the symbol table having a list of code points each uniquely identifying a symbol and a sort weight for the symbol identified by said each code point;

performing a binary search through each of a plurality of compression tables containing compressions for the

given language to find a matching compression that matches said combination of symbols in the input string, wherein the plurality of compression tables are searched in a descending order of compression types of the compression tables starting with a compression table having a compression type equal to said highest compression type for said first symbol.

12. (ORIGINAL) A computer-readable medium as in claim 1 1, wherein the compressions in each of the compression tables are sorted according to code points for symbols forming the compressions.

13. (ORIGINAL) A computer-readable medium as in claim 12, wherein each code point in the symbol table includes a tag indicating a highest compression type for said each code point, and wherein said step of referencing retrieves the tag for the code point identifying said first symbol.

14. (ORIGINAL) A computer-readable medium as in claim 13, wherein the tag for each code point in the symbol table is stored as a portion of the sort weight for said each code point.

15. (ORIGINAL) A computer-readable medium as in claim 11, wherein the code points in the symbol table are assigned to symbols according to the Unicode standard.

16. (ORIGINAL) A computer-readable medium as in claim 1 1, wherein the computer-executable instructions for performing a binary search form module that is called for searching each of the compression tables.

17. (CURRENTLY AMENDED) A computer-readable medium as in claim 1 1, having further computer-executable instructions for storing a ~~search~~ sort weight for the matching compression.

18. (ORIGINAL) A method of performing a linguistic sorting operation, comprising:

receiving an input string containing a plurality of linguistic symbols used in a given language;

for a first symbol in a combination of symbols in the input string, obtaining a highest compression type for compressions beginning with said first symbol;

performing a binary search through each of a plurality of compression tables containing compressions for the given language to find a matching compression that matches a combination of said first symbol and adjacent symbols in the input string, wherein the plurality of compression tables are searched in a descending order of compression types of the compression tables starting with a compression table having a compression type equal to said highest compression type for said first symbol.

19. (ORIGINAL) A method as in claim 18, wherein the step of obtaining the highest compression type includes referencing a symbol table that contains a list of code points each uniquely identifying a symbol and a sort weight for the symbol identified by said code point.

20. (ORIGINAL) A method as in claim 19, wherein the symbol table includes a tag for each code point indicating a highest compression type for said each code point, and wherein said step of obtaining retrieves the tag for the code point identifying said first symbol.

21. (ORIGINAL) A method as in claim 20, wherein the tag for each code point in the symbol table is stored as a portion of the sort weight for said each code point.

22. (ORIGINAL) A method as in claim 19, wherein the code points in the symbol table are assigned to symbols according to the Unicode standard.

23. (ORIGINAL) A method as in claim 18, where in the step of performing a binary search thorough each of the compression tables includes calling a search module to perform a binary search in each of the compression tables.

24. (ORIGINAL) A method as in claim 23, wherein the compressions in each of the compression tables are sorted according to code points for symbols forming the compressions, and wherein the binary search through each compression table is based on the code points for symbols forming the compressions in said each compression table.